Contents

1 Introduction

1.1 Motivation

1.1.1 Protein Structure

Proteins are essential molecules for living organisms. According to Watson (2021) it is approximated that a human body contains between 80,000 and 400,000 such proteins. They are involved in many different functions of the human body including, but not limited to, transporting oxygen in the blood, catalyzing chemical reactions and providing protection against pathogens. Each protein is made up of a unique sequence of amino acids. Various inexpensive and quick experimental methods have been devised over the years that can determine the amino acid sequence from the protein itself. The particular amino acid sequence determines the physiochemical properties of a protein which, in turn, determine how a protein folds and what 3D structure[1] it takes. The specific shape of a protein is related to its unique functionality.

Understanding the structure of proteins is of utmost importance. Designing drugs that can target particular proteins can aid drug discovery, engineering new proteins with desired properties can be used in biotechnology applications and understanding how proteins misfold and aggregate can help develop new treatments for diseases such as Parkinson's disease and Alzheimer's. Along with these, there are many other instances in which understanding a protein's structure can be of use.

Experimental methods for determining protein structures exist and can be highly accurate. However, they are associated with many challenges which provide an incentive to explore new methods for the task. Significant limitations of experimental methods include how time consuming and expensive they are. Additionally, some proteins are very difficult to study as they are too complex, hard to purify or too unstable. Therefore, new computational methods for protein structure prediction and understanding that can overcome these limitations are in high demand.

1.1.2 Protein Domains

A protein domain is a unit of a protein that folds and functions independently from the protein. Figure 1.1 shows an example of the structure of a protein highlighting its three domains. Protein domain boundaries are the residue positions that separate two domains.

The identification of these boundaries enables the division of proteins into smaller, independent domains. Subsequently, other methods can be applied to these domains, leading to better results such as in 3D structure prediction. Additionally, breaking down proteins into domains facilitates the process of crystallisation[2], which also aids in structure prediction using experimental methods.

[1] The atomic coordinates of a protein in 3D space

[2] A step in a process called X-ray crystallography which is an experimental method for determining the 3D structure of a protein

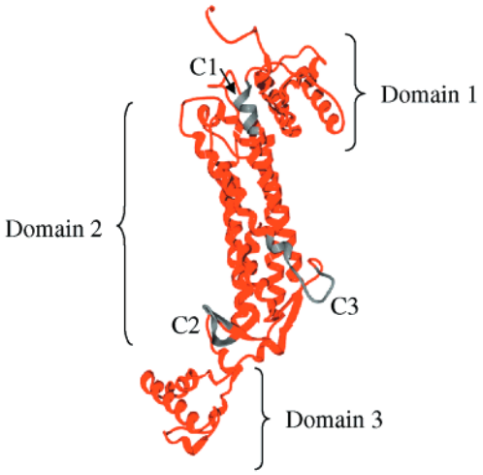

Figure 1.1: *A schematic diagram of the three dimensional topology of a protein indicating its three structural domains (Sampaleanu et al. 2001)*

1.2 Aims

1.2.1 Deep Learning

In light of the above motivations, we will explore how well a deep neural network can predict protein domain boundaries from the amino acid sequence. To achieve this we will:

- describe the problem of protein structure prediction from the amino acid sequence using machine learning based approaches.
- collect and process adequate data that will be used for training.
- implement and train a deep neural network with an architecture suited for the problem domain using different amino acid sequence encoding mechanisms.
- evaluate and analyse the model's results and compare them with other similar machine learning methods that exist in literature.
- discuss the limitations of our methods and suggest possible improvements that could have been adopted in our solution.

1.2.2 Evaluation

In chapter 2 we introduce and discuss a novel algorithm for predicting the 3D structure of a protein from its amino acid sequence, called AlphaFold. In light of its success at the 14th Critical Assessment of Structure Prediction[3], we aim to provide an insight into its efficacy in protein domain boundary prediction. The motivation for doing so is to argue whether domain boundary prediction could be facilitated in its process to improve its predictions. Our goals are to:

- describe the problem of evaluating AlphaFold for the domain boundary prediction task.
- collect appropriate AlphaFold-predicted 3D structures and translate them to AlphaFold-predicted domain boundaries.
- evaluate the results and compare them with results obtained from methods devised for explicitly predicting domain boundaries, including our model.
- discuss our findings and the limitations of our process.

[3]Community wide experiment to determine and advance the state of the art in modeling protein structure from amino acid sequence

2 Background

2.1 Proteins

2.1.1 Amino acid sequence

Amino acids are organic compounds that form the building blocks of proteins. Each is made up of different atoms and therefore has different properties. There are hundreds of unique amino acids found in nature but only about 20 are needed to make all the proteins found in the human body and most other forms of life (Lopez and Mohiuddin 2022). Table A.1 in Appendices presents the names of the 20 amino acids and their abbreviations.

2.1.2 Chains

2.1.3 Residues

Linkers

2.1.4 Domains

Continuous, Discontinuous

2.2 Recurrent neural networks

2.2.1 Classic recurrent neural networks

Recurrent neural networks (RNNs) is a class of neural networks suited for processing sequential data. The name "recurrent" comes from the fact that these networks include cycles in their architecture, allowing them to perform recurrent computations while taking past information as input and incorporating it into their current output, making them attractive for sequence-based predictions. However, they suffer from a well-known problem called gradient vanishing[1], rendering them incapable to model long-term dependencies. This limitation presents a motivation to investigate alternative architectures.

2.2.2 Long short-term memory

Hochreiter and Schmidhuber (1996) present the long short-term memory (LSTM) which is a special type of a recurrent neural network capable of addressing the challenge of long-term dependencies by partially[2] solving the vanishing gradient problem. LSTMs are capable of capturing long-term interactions during learning by controlling the flow of information within the network. This feature makes LSTMs more suitable than RNNs for longer sequences of data.

[1] A problem in deep learning where the gradient of the loss with respect to the current weight becomes vanishingly small, preventing or completely stopping the neural network from further training.

[2] The gradient can still vanish but not as rapidly as it does in classic RNNs

2.2.3 Bidirectionality

Bidirectional RNNs (Bi-RNNs) process a sequence of inputs in both forward and backward directions, enabling the network to learn from both past and future inputs concurrently. This addresses the limitation of RNNs, and in turn of LSTMs, which only incorporate past information into the current output, neglecting the potential usefulness of future information in prediction.

2.3 Sequence encoding

2.3.1 One-hot

One-hot encoding is a process for converting categorical data to binary vectors which can be used as training data in machine learning. Each binary vector is filled with zeros in all indices except the index that corresponds to the category being represented. In the case of converting amino acids into binary vectors, since there are 20 unique amino acids, each is represented by a binary vector of length 20 with zeros everywhere except the i_{th} position if the amino acid is the i_{th} amino acid in the list of unique amino acids.

2.3.2 Evolutionary Scale Modeling (ESM)

Rives et al. (2020) have trained an unsupervised deep neural network based on the Transformer (Vaswani et al. 2017) architecture, on 86 billion amino acids to learn protein representations. It is claimed that these representations produce features that generalize across a range of applications including structure prediction. A number of pre-trained[3] models from this study are available to the public and have been extensively used in recent protein domain and domain boundary prediction methods. (is more detail needed?)

2.3.3 Convolutional autoencoding representations of proteins

Yang et al. (2023) point out that recent models successful in learning protein representations, such as ESM, rely on the Transformer architecture which scales quadratically with sequence length in both run-time and memory. Incentivised by this limitation, they have trained a more efficient neural network, namely a convolutional autoencoder, (is it?) on the masked language model pre-training task[4] and show that their results are competitive or superior than Transfomers in downstream applications such as structure prediction. Convolutional autoencoders use convolutions that scale linearly - instead of quadratically - and are able to "inherently incorporate relative positional information, since sequences are modeled as sliding windows of amino acids". This study is more recent than ESM and has not been mentioned in any literature on protein domain boundary prediction. (more info maybe)

(check the methodology, are representations amino-acid based or sequence based?)

2.4 Scoring metrics

2.4.1 F1

Compare MCC with F1

[3] Models which their weights do not need to be optimised as the model has already been trained.
[4] explain what this is

2.4.2 Mathew's correlation coefficient

Advantages of MCC over F1: Chicco and Jurman (2020).

2.4.3 Domain boundary distance (DBD

2.4.4 Normalized domain overlap (NDO)

2.5 Protein Domain Prediction

Wang et al. (2022) explored and evaluated the performance of a novel algorithm , namely Res-Dom, for protein domain boundary prediction. They extract four features from the amino acid sequence: solvent accessibility, secondary structure, a Hidden Markov model profile and the sequence's embedded features from a pre-trained protein language model. Then, a Residual Neural Network, followed by a bi-directional long short-term memory (BLSTM), take these four features as input and output the predicted domain boundaries. Their evaluation shows that this method can achieve highly accurate results. The limitation of this method is that it depends on other methods for extracting features from the amino acid sequence. For example, the work of Cheng et al. (2005) used to extract the secondary structure and solvent accessibility can only work on proteins with a sequence length with less than 1500 residues. In turn, this limitation is also inherited by Res-Dom for predicting protein domain boundaries.

Shi et al. (2019) present a method for predicting domain boundaries and argue that local and non-local interactions between residues provide useful information for the overall structure of the protein. This is taken into account and a method that captures both local and non-local interactions has been devised using a multi-channel Convolutional Neural Network (CNN) and a stacked bidirectional Gated Recurrent Unit (BGRU) In doing so, the work of Cheng et al. (2005) is again present which also introduces the limitation on the number of residues for a sequence.

Jiang et al. (2019) have followed an *ab initio* approach without using third party methods which can introduce limitations such as the one seen in Res-Dom or DNN-Dom. This method only depends on the amino acid sequence of proteins which is used to train a stacked BLSTM. Each amino acid sequence was encoded using 5 numerical descriptors collected from a comprehensive list compiled from public databases. Their results demonstrate that, although this method is faster than template-based methods and has no dependencies, the accuracy is lower.

Postic et al. (2017) have taken a structure-based approach. That is, their method predicts domain boundaries from the experimental structures of proteins. During the evaluation stage of their research, they have tested the accuracy of their method both with and without an 85% boundary overlap criterion[5] and have concluded that finding the correct number of domains is a more challenging problem than delimiting accurate boundaries.

The work of Zheng et al. (2020) offers its own contribution in protein structure prediction by predicting the domain each residue belongs to which provides more information about the structure of the protein than the domain boundaries do.

2.6 AlphaFold

AlphaFold 2 (AF2), the work of Jumper et al. (2021), is the most successful and accurate method to date for predicting the entire 3D structure of a protein according to the 14th Crticial Assessment of Structural Prediction. The network consists of two main components. The first utilizes the work of Vaswani et al. (2017) to generate contextual representations from the input data: a

[5] Explain what this is

contact map, which captures which residues interact with each other, and representation of the chain's homologous sequences. The second is responsible for generating the 3D structure of the protein utilizing the generated representations, and by utilizing an ensemble of multiple prediction models to improve its accuracy. Even though AF2 demonstrates promising capabilities, its practical use is under exploration, including protein domain boundary assignment.

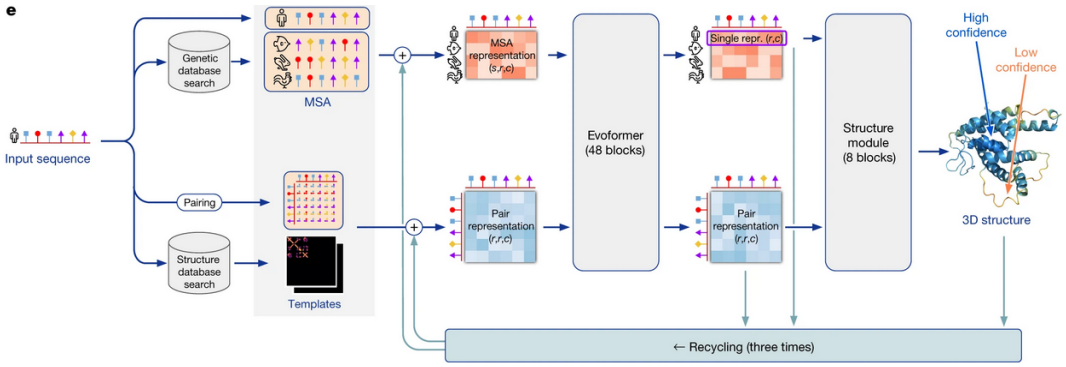

Figure 2.1: *AlphaFold*

2.7 Summary

In this chapter we discussed what proteins are and defined the necessary terminology that will be mentioned throughout the dissertation. We introduced RNNs and RNN variants that solve technical challenges and improve learning. We have presented related work in the field of protein domain boundary prediction that shows that deep learning can be an effective and low-cost method to predict structural information of proteins, yet, there is still room for improvement and testing. Finally we introduced AlphaFold, gave a high-level abstract explanation of its architecture and discussed its recent success in 3D structure prediction.

3 Analysis/Requirements

What is the problem that you want to solve, and how did you arrive at it?

3.1 Guidance

Make it clear how you derived the constrained form of your problem via a clear and logical process.

The analysis chapter explains the process by which you arrive at a concrete design. In software engineering projects, this will include a statement of the requirement capture process and the derived requirements.

In research projects, it will involve developing a design drawing on the work established in the background, and stating how the space of possible projects was sensibly narrowed down to what you have done.

3.2 Predicting Domain Boundaries

We assume that there is a pattern in every protein sequence from which a machine learning algorithm can learn to predict the domain boundaries. This is evident from two facts:

- The literature discussed in Chapter 2 shows that accurate results are achievable.
- The thermodynamic hypothesis (Anfinsen 1973) states that "the native structure is determined only by the protein's amino acid sequence". Since the domain boundaries are caused by the structure of the protein, the amino acid sequence also determines the domain boundaries.

CATH is a publicly available database from where we can obtain the true[1] domain boundaries for a large number of proteins. This information can be utilized to train a deep neural network that can capture the complexity of the link between sequences of amino acids and domain boundaries.

The available data for this task include the 3D coordinates of the atoms in the protein and the amino acid sequence.

3.3 Protein Representations

Many recent machine learning methods for protein domain boundary prediction use the ESM model to encode protein sequences. Wang et al. (2022) have shown that, when using ESM, their model scores higher in most metrics used to measure their model's performance. Motivated by the recent release of CARP (Yang et al. 2023), we will train a neural network using ESM and one using CARP and compare their performance.

[1] As determined by experimental methods which are highly accurate and the best available option in terms of accuracy for gathering structural information of proteins.

Note that, the comparison will be made between versions of the models with a similar number of trainable parameters (35 million for ESM and 38 million for CARP) to make the comparison as fair as possible. The state-of-the-art models for each require enormous computational resources to use, as they have over 600 million trainable parameters each, and we therefore are not able to use.

3.3.1 Discontinuous domains

Discontinuous domains are domains that contain more than one fragment from different regions of the sequence. Figure (?) shows an example of a discontinuous domain. Therefore, in the presence of discontinuous domains the assumption that the number of domains N_d is equal to the number of domain boundaries $N_b + 1$ fails. Methods for predicting domain boundaries are still very inaccurate and methods do not attempt to solve it such as (example 1) and (example 2) from Chapter 2. Our metrics for NDO and DBD will therefore only consider proteins without discontinuities as we train the neural network to learn where domain boundaries lie (and in turn learn the number of domains only for non-discontinuous proteins) and not to which domain each amino acid/residue belongs to. (but we do include discontinuous proteins during learning as that does not affect the task of learning domain boundaries).

3.4 Evaluating AlphaFold

We aim to evaluate the performance of AlphaFold in the task of domain boundary prediction and compare it both with available literature and with our model. Because AlphaFold is designed to predict the 3D structure of the protein from the sequence and not the domains, we will translate the AlphaFold predictions into protein domain boundaries and evaluate the predictions using various metrics . By evaluating AlphaFold's performance, we can gain a better understanding of how it performs in this specific task and identify any potential room for improvement by utilizing protein domain boundary information in the AlphaFold algorithm.

Furthermore, we are intersted in how AlphaFold can perform in the task domain boundary prediction particularly on discontinuous domains which still remains of high interest and importance in literature.

3.5 Summary

In this chapter, we described the problem of predicting protein domain boundaries from the amino acid sequence, and the process by which we arrived at this problem. We also explain how we plan to tackle this problem using machine learning techniques, and evaluate the performance of AlphaFold in this task.

4 Design

How is this problem to be approached, without reference to specific implementation details?

4.1 Guidance

Design should cover the abstract design in such a way that someone else might be able to do what you did, but with a different language or library or tool. This might include overall system architecture diagrams, user interface designs (wireframes/personas/etc.), protocol specifications, algorithms, data set design choices, among others. Specific languages, technical choices, libraries and such like should not usually appear in the design. These are implementation details.

4.2 Description of the machine learning system

4.2.1 Overview

The system will take sequences of amino acids of variable length L and will predict a probability vector of length L that assigns a probability to every position in the sequence, indicating the probability that the residue/amino acid** is a domain boundary. These probabilities will be computed by the neural network which learned what probabilities yield the best evaluation metrics. For the neural network to learn this task, we will compare the probability vector with the ground truth and calculate the error between the two. The error will be passed to the neural network to adjust its parameters accordingly in order to improve its predictions.

4.2.2 Input sequence

The machine learning algorithm cannot take a sequence of characters as input. Therefore, we will encode each sequence using an encoding mechanism that generates a representation of each sequence. The encoding mechanism will take an amino acid sequence of length L and will output a matrix of shape $L \times N$ where N is the number of features per amino acid.

In Chapter 2 we introduced three different encoding mechanisms. We will use each independently and compare the performance of our neural network when using each of the encoding techniques in Chapter 6.

4.2.3 Ground truth

The ground truth for every amino acid sequence is a binary vector of length L with zeros everywhere except at every index i if the i^{th} amino acid is considered a domain boundary. (maybe this must go in the implementation) However, we find it difficult for the neural network to learn the exact position of boundaries. Therefore, we amplify the signal of each boundary by assigning the neighboring residues/amino acids** of every boundary the value of 1, creating regions where a domain boundary is very likely to exist. The resulting vector is what the prediction of the network is compared to during training.

During testing, we take the median residue/amino acid of every region as the domain boundary and use that to evaluate our metrics.

4.2.4 Choice of cut-off threshold

The neural network produces a probability vector, and the ground truth is a binary vector. To convert the probabilities into binary predictions, we choose a threshold t. Probabilities above t are classified as 1, and probabilities below (or equal to) t are classified as 0. We optimize the choice of t by maximizing the Mathew's Correlation Coefficient.

4.2.5 Data/Data Collection

Converting domain boundaries to a binary vector. IID random variables. Balanced dataset.

4.3 AlphaFold evaluation system design

The system will take a PDB code as input which will be mapped to a UniProt accession[1]. The PDB code and the accession of the AlphaFold-predicted structure will be used to fetch their respective PDB files[2]. Because the PDB files only contain the 3D coordinate of the atoms in the protein, we must explicity assign the domains in the structure. To do this, we will use a domain assignment software that parses a PDF file and computes the position of each domain in the protein. This information will be used to extract the domain boundaries and compare them with the ground truth that is available on the CATH database.

Note that, because the domain assignment is not perfect, we feed the true PDB file in the algorithm so get an upper bound of that is the best result that can be obtained.

Figure 4.1 is a high level abstract diagram that shows how the evaluation of AF will take place.

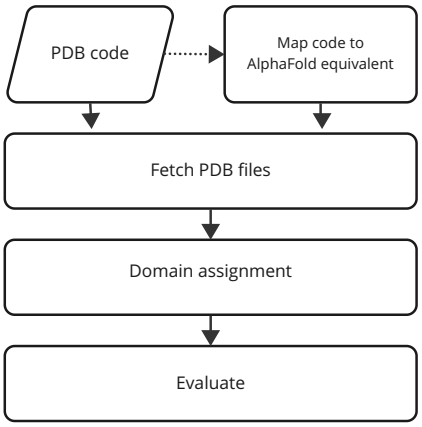

Figure 4.1: *Flow chart of the AlphaFold evaluation system design. Starting with a PDB code, we will map that to the code of the AlphaFold-predicted structure. Then both codes are used to fetch the respective PDB files that contain the structural information. A method for assigning domains from the PDF file is then used to find the domain boundaries, which are then compared to the ground truth. Finally we compare and evaluate the results obtained from both PDB files.*

[1] The format of the unique identifier that AlphaFold is using for its database entries.
[2] The file with the structural information of a protein.

4.4 Summary

We have examined how the system will learn to predict domain boundaries. Figure 4.2 is a high-level abstract diagram that shows how the data flows in the system. We have discussed input sequence, ground truth (this may go), the choice of cut-off threshold and how the data is collected.

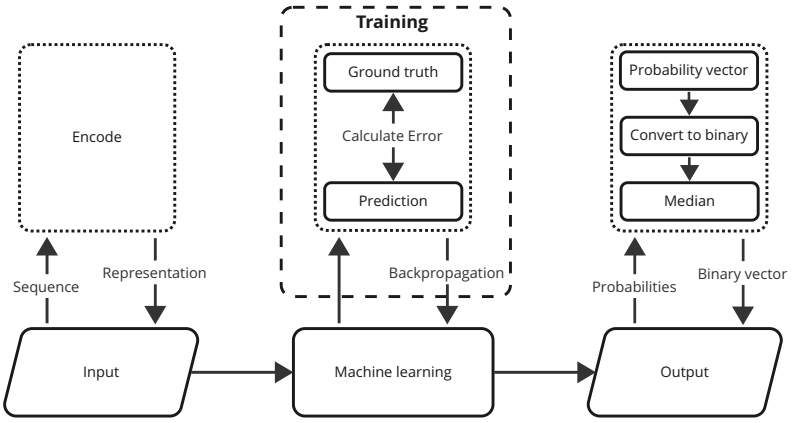

Figure 4.2: *Flowchart of the machine learning system design. Starting with the input sequence, it is given as input to the encoding mechanism that produces a representation matrix. During training the parameters of the neural network are optimised to minimise the error between the prediction and the amplified signal that comes from the ground truth. When not in training mode, the system*

5 Implementation

What did you do to implement this idea, and what technical achievements did you make?

5.1 Guidance

You can't talk about everything. Cover the high level first, then cover important, relevant or impressive details.

5.2 System overview

We have implemented a deep neural network model which uses the amino acid sequences of proteins to make predictions about the domain boundaries of the proteins. We have collected and processed adequate data from public databases and trained the neural network on those. We have also implemented a system which translates the AlphaFold-predicted 3D structure of proteins into protein domain boundaries in order to evaluate AlphaFold's performance on the task.

5.3 Data collection and processing

- Stratified sampling
- Independent and identically distributed random variables
- Balancing

5.3.1 Overview

To train the neural network we need to create a dataset with amino acid sequences of proteins coupled with their domain boundaries. We create this dataset using a combination of two publicky available databases: RCSB PDB and CATH. RCSB PDB provides `pdb-code.pdb` files for a large number of proteins where each contains the amino acid sequence of the protein along with the 3D structure of the protein which has been determined using experimental methods. CATH provides a list of PDB codes coupled with their domain boundaries.

Example of CATH format:

- 10gsA01 2-78,187-208
- 10gsA02 79-186

5.3.2 Sampling

We first started with sampling chains from the CATH list. We used stratified sampling to collect a random sample from chains of different sizes in terms of their number of domains. To do this, we parsed the entire list and put each chain into a category where each category is characterised by the number of domains the chains in the category have. Figure ?? shows the distribution of chains with n domains

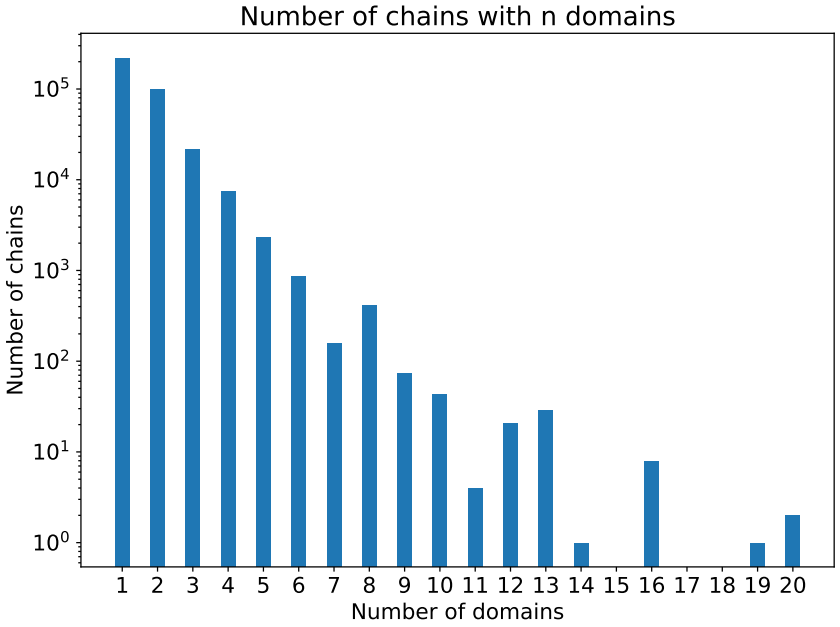

Figure 5.1: *Distribution of chains with regards to the number of domains. Note that the scale of the y-axis is logarithmic.*

The number of domains a protein has is directly proportional to the number of boundaries. Therefore, we aimed to create a more balanced dataset in order to avoid biases from being introduced to our model.

5.3.3 Sequence similarity

After performing stratified sampling, we

5.4 Machine learning

- Architecture
- Training, loss
- hardware, time to train
- post processing, median

5.5 Metrics

- Cut-off optimisation for MCC
- Implementation of MCC, NDO, DBD, sliding window

5.6 AlphaFold

- Domain assignment
- collecting the appropriate PDB UniProt pairs